DTIC FILE COPY
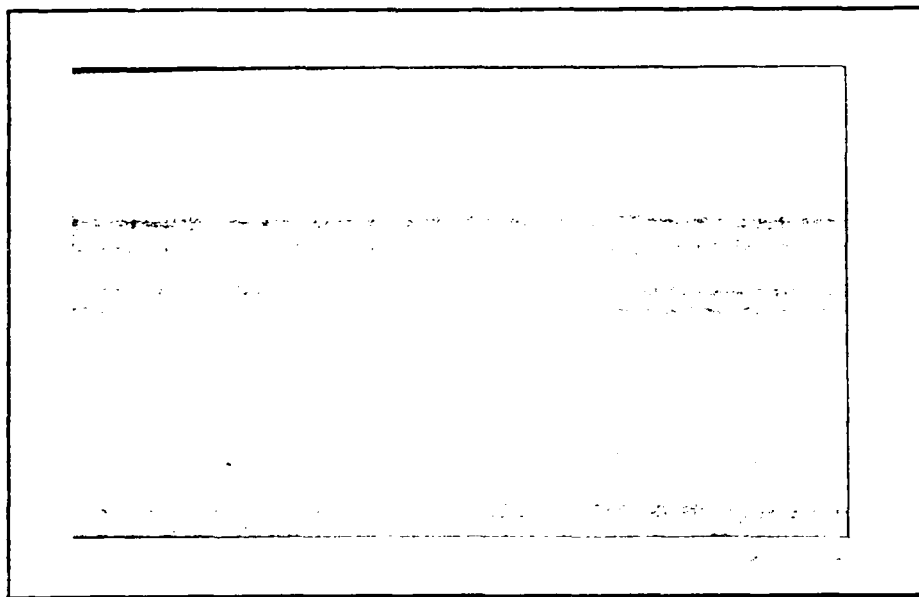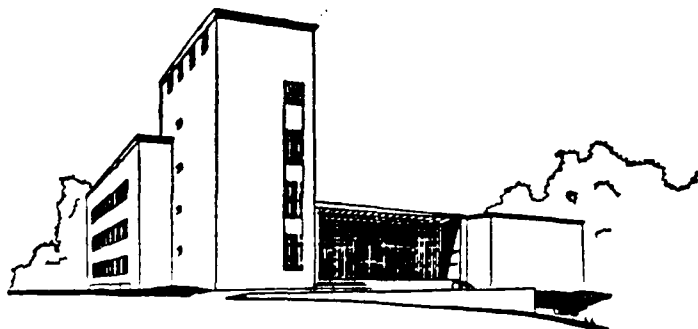
Carnegie Mellon University

PITTSBURGH, PENNSYLVANIA 15213

GRADUATE SCHOOL OF INDUSTRIAL ADMINISTRATION

WILLIAM LARIMER MELLON, FOUNDER

DTIC
ELECTE
MAY 0 9 1988
S   E   D

88   5   06   136

ABSTRACT

This paper compares approaches proposed in the literature for the Capacitated Plant Location Problem. The comparison is based on new theoretical and computational results. The main emphasis of the paper is on relaxations. In particular we identify dominance relations among the various relaxations found in the literature. We also perform a probabilistic study of these relaxations using a Euclidean model. In the computational study, we compare the relaxations as a function of various characteristics of the test problems. Several of these relaxations can be used to generate heuristic feasible solutions that are better than the classical greedy or interchange-type heuristics, both in computing time and in the quality of the solutions found. (Keywords: plants; warehouses; mixed integer linear program).

1. Introduction

For many organizations the location of plants, such as warehouses or factories, affects the cost of supplying commodities to clients through the transportation costs on the one hand and the fixed costs of opening and operating the plants on the other hand. These costs vary with the location and size of the plants. In this paper we study a classical model in location theory.

The Capacitated Plant Location Problem (CPLP) can be formulated as the mixed integer linear program

$$Z = \text{Min } \Sigma_i \Sigma_j \, c_{ij} x_{ij} + \Sigma_j \, f_j y_j$$

(D) $\qquad \Sigma_j \, x_{ij} = 1 \qquad$ for all i

(C) $\qquad \Sigma_i \, d_i x_{ij} \leq s_j y_j \quad$ for all j

(N) $\qquad 0 \leq x_{ij} \leq 1, \; 0 \leq y_j \leq 1 \;$ for all i and j

(I) $\qquad y_j \;$ integer $\qquad$ for all j,

where d_i is the demand of client i, f_j and s_j are the operating cost and capacity of facility j, if it is open, and c_{ij}/d_i is the unit transportation cost between i and j. The decisions to be made are represented by the variables y_j and x_{ij}, namely $y_j = 1$ if facility j is open and 0 otherwise, and x_{ij} is the fraction of the demand of client i met from facility j. The constraints (D) are the demand constraints, (C) the capacity constraints, (N)

the nonnegativity and simple upper bound constraints and (I) the integrality constraints.

The applications of this model are not limited to plant location. For example, the same mathematical model is appropriate for making optimal lot sizing decisions in production planning. If d_i is the demand in period i, f_j and s_j are the set up cost and production capacity in period j, then CPLP is a valid model where the decision variables are interpreted as follows: x_{ij} is the fraction of the demand d_i produced in period j and $y_j = 1$ if production occurs in period j, 0 otherwise. Finally, in this model, c_{ij}/d_i is the unit production cost in period j plus the inventory holding cost until period i, if $i > j$ (or the backlogging cost if $i < j$). The model CPLP has also been proposed in the context of telecommunication network design (Kochman and McCallum [22] and Mirzaian [24]), machine replacement and vehicle routing (see Sridharan [28]).

The formulation of the CPLP can be enriched by simple valid inequalities. For example, the variable upper bound constraints

(B) $\quad x_{ij} \le y_j \quad$ for all i,j

and the total demand constraint

(T) $\quad \Sigma_j s_j y_j \ge \Sigma_i d_i$.

To see that (B) is valid for CPLP, note that $x_{ij} \le 1$ by constraint (N). Using (C) and (I) it follows that $x_{ij} \le y_j$. To see that (T) is valid. sum up the constraints (C) and use (D).

One obtains various lower bounds on Z by relaxing subsets of the constraints (D), (C), (I), (B) and (T) either completely or in a Lagrangian fashion. We use the following notation. If a set of constraints, say (A), is relaxed completely the resulting bound is denoted by Z^A. If constraints other than (I) are relaxed in a Lagrangian fashion, the corresponding Lagrangian dual bound is denoted by Z_A. For example

$$Z_D = \underset{u}{\text{Max}} \underset{x,y}{\text{Min}} \; \Sigma_i \Sigma_j \; c_{ij} x_{ij} + \Sigma_j \; f_j y_j - \Sigma_i \; u_i (\Sigma_j \; x_{ij} - 1)$$

$$\Sigma_i \; d_i x_{ij} \leq s_j y_j \qquad \text{for all } j$$

$$0 \leq x_{ij} \leq y_j \qquad \text{for all } i,j$$

$$\Sigma_j \; s_j y_j \geq \Sigma_i d_i$$

$$y_j \; \varepsilon \; \{0,1\} \qquad \text{for all } j.$$

These lower bounds provide the basis for numerous algorithms. For example, the following relaxations have been used in branch and bound algorithms.

Z^{BI} (This relaxation is the so-called weak linear programming relaxation of CPLP. It was used by Sa [27] and improved by Akinc and Khumawala [2] using ad hoc rules).

Z^I (This is the strong linear programming relaxation of the problem. It was used by Davis and Ray [7], Guignard and Spielberg [18], and Baker [3])

Z^T_D (Geoffrion and Mc Bride [14])

Z_D (Nauss[25])

Z'_D (Christofides and Beasley [6]. In this relaxation, the total demand constraint (T) is replaced by $\Sigma_j \, y_j \geq K$ where K is the smallest integer such that the sum of the K largest s_j's is at least $\Sigma_i d_i$).

Z_C (Van Roy [30] : Guignard and Kim [16]).

Recently, a new type of relaxation has been introduced by several authors. The method, known as <u>variable splitting</u>, can be defined in general for (mixed) integer programs. It is obtained by rewriting the program

$$Z = \min c \, u$$

$$(S_1) \qquad A_1 \, u \leq b_1$$

$$(S_2) \qquad A_2 \, u \leq b_2$$

$$u \in X$$

as

$$Z = \min c \, u$$

$$A_1 \, u \leq b_1$$

$$u \in X$$

$$A_2 \, \upsilon \leq b_2$$

$$\upsilon \in X$$

$$u = \upsilon$$

and by performing a Lagrangian relaxation of the equalities $u = \upsilon$. The Lagrangian dual bound obtained in this way has the value

$$Z \, (S_1, \, S_2) = \min c \, u$$

$$u \in \{A_1 u \leq b_1, \, u \in X\} \cap \{A_2 u \leq b_2, \, u \in X\}$$

whereas the classical Lagrangian bound obtained by dualizing (S_1) is

$$Z_{S_1} = \min cu$$

$$u\epsilon \{A_1 u \le b_1\} \cap \{A_2 u \le b_2, u\epsilon X\}$$

(see Guignard and Kim [17] and Geoffrion [13] respectively). As a consequence, $Z_{S_1} \le Z(S_1, S_2)$, i.e. the bound obtained by variable splitting is always as tight as the corresponding Lagrangian bound.

In a recent paper, Barcelo, Fernandez and Jönsten [4] propose the following split for CPLP. (S_1) corresponds to the demand constraints (D), X to the nonnegativity and 0, 1 requirements (N) and (I), and (S_2) to the other constraints, namely (C), (B) and (T). We show in Section 2 that the proposed relaxation yields the bound Z_D. Other splits yield other relaxations.

Our motivation for this study is twofold, theoretical and computational. On the theoretical side, we rank relaxations according to the tightness of the bounds that they yield. This analysis is performed in section 2. Several inequalities given in that section are classical, but we have had a few surprises. For example, consider Z_C, Z_{TC} and Z^T_C. It is easy to show that $Z_C \ge Z_{TC} \ge Z^T_C$ and both Van Roy [30] and Guignard and Kim [16] have noticed that the bound Z_C is often strictly stronger that Z^T_C in practice. However Z_C is also harder to compute than Z^T_C and, in both references, the Lagrangian relaxation of (T) is introduced. Indeed, it seems that computing Z_{TC} is a good compromise between bound quality and ease of solution. Our Theorem of Section 2 shows that, in fact, $Z_{TC} = Z^T_C$. So Z_{TC}

cannot be used as a shortcut in approaching the quality of the bound Z_C. On the other hand, the solution of Z_{TC} by subgradient optimization may still have merits as it is possible that it converges faster to the common bound $Z_{TC} = Z_C^T$ than a subgradient approach applied directly to Z_C^T.

Another interesting result from Section 2 is that variable splitting does not yield stronger bounds than Z_C.

Section 3 analyzes the complexity of solving the different Lagrangian relaxations. We also consider the complexity of the problem CPLP itself under two different assumptions on the capacities of the facilities.

Section 4 contains our computational experience with various relaxations. Nearly all the researchers in the field have tested their algorithms on the same set of data, the famous Kuehn and Hamburger test problems as adapted by Khumawala [21]. These test problems have been a great help in comparing the various solution methods. At the same time, it should be noted that this test set is relatively small and has a very special property : all the facilities have the same capacity and operating cost. Therefore it seems appropriate to conduct a computational study on a different and larger set of test problems. These problems have varying capacities and operating costs but, for each facility, the operating cost is correlated to the capacity according to an increasing concave function (economies of scale). The duality gap yielded by various relaxations appears to be sensitive to characteristics of the data such as total capacity versus total demand. To demonstrate such relationships we perform our computational experiments as a function of $\rho = \Sigma_j s_j / \Sigma_i d_i$. Keeping everything else equal, we scale the s_j's in order to get different values of ρ. For example we find that adding the total demand constraint (T) improves the relaxations significantly when ρ is small.

Our computational experience also covers some existing heuristics. Here as well we have had surprises. A number of heuristics which perform well on Kuehn and Hamburger problems have a poor performance on our test problems. We recommend the use of relaxation-based heuristics rather than the usual greedy or interchange-type heuristics. Consider any iterative method (such as subgradient optimization) to compute Z_D. For each u^t, $t \geq 1$, generated during the iterative process, the subproblem in x and y

$$\text{Min}_{x,y} \ \Sigma_i \Sigma_j \ c_{ij} x_{ij} + \Sigma_j \ f_j y_j - \Sigma_i \ u_i (\Sigma_j \ x_{ij} - 1)$$

$$\Sigma_i \ d_i x_{ij} \leq s_j y_j \qquad \text{for all } j$$

$$0 \leq x_{ij} \leq y_j \qquad \text{for all } i, j$$

$$\Sigma_j \ s_j y_j \geq \Sigma_i d_i$$

$$y_j \ \epsilon \ \{0,1\} \qquad \text{for all } j$$

is solved. The vector (x_D, y_D) so obtained produces a Lagrangian bound, but the vector y_D can also be used to generate a heuristic solution. Indeed, given y_D, one can solve a transportation problem to get a feasible solution $\bar{x}_D$. Thus at each iteration of the iterative process, a heuristic solution $(\bar{x}_D, y_D)$ can be generated. The best solution found in 50 or so subgradient iterations can be used as a heuristic. We find that this heuristic is just as fast and gives much better solutions than the more commonly used greedy-interchange approach. Instead of Z_D, one can also use Z_C^T or Z_C as the basis for the heuristic but we do not recommend it, as little additional accuracy is gained at much computational cost.

Section 5 contains probabilistic results. This analysis improves our understanding of the CPLP in a new dimension and complements the computational study of Section 4. For the probabilistic model that we choose, the relaxations fall into three categories. For relaxations such as Z^{BI}, the relative error as an estimate of Z can be expected to be over 66%. For bounds such as Z^I or Z_D, the relative error can be expected to be below 0.2%. Finally for the relaxations Z_C, Z_{CT} and Z_C^T, the relative error can be expected to go to 0 as the size of the instances goes to infinity. In particular, we show that the classical linear programming relaxation Z^I has a "duality gap" which is no greater for CPLP than for the Simple Plant Location Problem (SPLP). This seems to contradict the common belief that CPLP's have larger duality gaps. We should remember that our probabilistic analysis is limited by various assumptions. For example, it assumes that the number of clients served by each open facility is large.

2. Relative Strength of the Relaxations

In Theorem 1, we compare the strength of the various linear programming relaxations and lagrangian duals of CPLP. We compare all the possible bounds that can arise in this way, except that we do not relax (N) (there is never any computational gain in doing so), nor do we consider complete relaxations of (C) or (D) since we feel that such bounds would be of little interest in solving CPLP (Z^C and Z^D are NP-hard to compute, yet they lack important aspects of CPLP.) Specifically, we consider all the combinations where (T) and (B) are present, relaxed completely or in a Lagrangian fashion, and where (C) and (D) are present or relaxed in a Lagrangian fashion, i.e. 3x3x2x2 = 36 relaxations, as well as the LP relaxations Z^I, Z^{BI}, Z^{TI} and Z^{TBI}. In addition, we consider the relaxation Z_D' of Christofides and Beasley defined in

Section 1 above. Theorem 1 states that these 41 relaxations only yield 7 genuinely different bounds. It is interesting to note that 6 of these 7 bounds have already all been proposed in the literature as the basis for branch-and-bound algorithms.

In Theorem 2, we briefly discuss relaxations that can be obtained by variable splitting. We show that no new bound arises from these relaxations.

We use the following notation. Given a constraint set (S), P(S) denotes the convex hull of the vectors satisfying (S). More generally, if $(S_1),\ldots,(S_k)$ are constraint sets, $P(S_1\ldots S_k)$ denotes the convex hull of the vectors satisfying $(S_1),\ldots,(S_k)$. For example, with this notation, we can write

$$Z^{TB} = \min \; cx + fy$$

$$(x,y) \; \epsilon \; P(DCIN).$$

Using a classical theorem (see Geoffrion [13, Theorem 1D] for example), the Lagrangian duals can also be expressed using this notation. As an example,

$$Z_{TC} = \min \; cx + fy$$

$$(x,y) \; \epsilon \; P(TC) \cap P(DINB).$$

Now we are ready to prove the following Theorem.

Theorem 1 The following inequalities hold

$$(2.1) \quad z^{BI} \begin{cases} \leq \\ \leq \end{cases} z^I \begin{cases} \leq \\ \leq \\ z_C^B \end{cases} z_D' \begin{cases} \leq \\ \leq \\ \leq \end{cases} z_D \begin{cases} \leq \\ \leq \end{cases} z_C \leq z,$$

every inequality in (2.1) is strict for at least one instance of CPLP, and no other inequality exists among the values of (2.1) except those derived by transitivity.

The following equalities hold

$$(2.2) \quad z = z_B = z^B = z_T = z^T = z_{TB} = z_T^B = z_B^T = z^{TB}$$

$$(2.3) \quad z_D = z_{DC} = z_{BD} = z_{BC} = z_{BDC} = z_D^B$$

$$(2.4) \quad z_C^B = z_{DC}^B$$

$$(2.5) \quad z_C^T = z_{TC}$$

$$(2.6) \quad z^I = z^{TI} = z_{TD} = z_D^T = z_{TDC} = z_{DC}^T = z_{BTC} = z_{BC}^T = z_{BTDC} = z_{BDC}^T$$

$$= z_{TD}^B = z_D^{BT} = z_{BTD} = z_{BD}^T$$

$$(2.7) \quad z^{BI} = z^{TBI} = z_{TC}^B = z_C^{TB} = z_{TDC}^B = z_{DC}^{TB}.$$

Proof: In (2.1), the inequalities $z^{BI} \leq z^I$ and $z_C^T \leq z_C \leq z$ are immediate by relaxation. $z_D' \leq z_D$ follows from the fact that the constraint used in z'_D in place of (T) is actually implied by (I) and (T). Using the property that Lagrangian duals are at least as strong as the corresponding linear

programming relaxation (Geoffrion [13]), we get $z^I \leq z_D'$, $z^{BI} \leq z_C^B$ and $z^{IT} \leq z_C^T$. The last inequality implies $z^I \leq z_C^T$ provided we can show $z^I = z^{IT}$. This will be done in (2.6). $z_C^B \leq Z_D$ follows from $z_{DC}^B \leq Z_D$ provided we can show $z_C^B = z_{DC}^B$. This will be done in (2.4). Finally $Z_D \leq Z_C$ follows from $Z_{DC} \leq Z_C$ provided we can show $Z_D = Z_{DC}$. This will be done in (2.3). So, to show the validity of the theorem, it remains to prove the equalities (2.2) - (2.7) and to exhibit instances showing that all the inequalities in (2.1) can be strict and that no other inequality exists among the bounds considered in (2.1). These results are provided in the next two lemmas. □

Lemma 1 The equalities (2.2) - (2.7) hold.

Proof: We give polyhedral proofs of these results. See [28] for algebraic proofs.

The constraints (T) and (B) can be derived from (D), (C), (I) and (N) as shown in the introduction. Therefore P(DCIN) = P(TBDCIN), proving (2.2).

To prove (2.3), we first show that $Z_D = Z_{BDC}$ and then that $Z_D = Z_D^B$. The first equality holds if we can prove the polyhedral result

(2.8)   P(D) ∩ P(BCINT) = P(BDC) ∩ P(INT).

Since P(BDC) = P(D) ∩ P(BC) we only have to show

Claim

(2.9)   P(BC) ∩ P(INT) = P(BCINT).

Proof of Claim:

By definition of P, $P(BCINT) \subseteq P(BC) \cap P(INT)$. Now consider any extreme point of $P(BC) \cap P(INT)$, say $(\bar{x}, \bar{y})$. If $\bar{y}$ has only 0,1 coordinates, then $(\bar{x}, \bar{y}) \epsilon P(BCINT)$ and the claim is proved. So assume that $\bar{y}$ has at least one fractional component. We will show that this cannot occur. Denote by $P_y(INT)$ the projection of the polyhedron $P(INT)$ onto the y-space. Since $P_y(INT)$ has only integral vertices, $\bar{y}$ is not a vertex and therefore $\bar{y} = \frac{1}{2}y^1 + \frac{1}{2}y^2$ where $y^i, y^2 \epsilon P_y(INT)$. If $\bar{y}_j = 0$, set $x_{ij}^1 = x_{ij}^2 = 0$. Else, set $x_{ij}^k = \bar{x}_{ij} (y_j^k / \bar{y}_j)$ for k=1,2. We have $(x^k, y^k) \epsilon P(BC) \cap P(INT)$ for k=1,2 and $(\bar{x}, \bar{y}) = \frac{1}{2}(x^1, y^1) + \frac{1}{2}(x^2, y^2)$. This contradicts the assumption that $(\bar{x}, \bar{y})$ is an extreme point of $P(BC) \cap P(INT)$. Thus the claim is proved.

Now we prove $Z_D = Z_D^B$ by showing

(2.10) $\quad P(D) \cap P(BCINT) = P(D) \cap P(CINT)$.

This relation follows from the fact that (C), (I) and (N) imply (B). This completes the proof of (2.3).

To prove (2.4), we show that

(2.11) $\quad P(C) \cap P(INTD) = P(DC) \cap P(INT)$.

Since $P(DC) = P(C) \cap P(D)$, it suffices to show $P(D) \cap P(INT) = P(DINT)$. The constraints (D), (I), (N) and (T) completely separate into the constraints involving x and those involving y. The convexification step only involves the

variables y and, therefore, the result follows. This completes the proof of (2.4).

The equality (2.5) follows from $P(C) \cap P(DINB) = P(TC) \cap P(DINB)$, since (T) is a consequence of (C) and (D).

Now we show (2.6). First, $Z^I = Z^{TI}$ since (T) follows from (C) and (D). First, we give a polyhedral proof of $Z_D^{BT} = Z_{TD}$. We have $P(D) \cap P(CIN) = P(D) \cap P(CINB) = P(TD) \cap P(CINB)$, where the first equality holds because (C), (I), (N) imply (B) and the second equality holds because (D) and (C) imply (T). So all the relaxations in (2.6) are at least as strong as $Z^I = Z^{TI}$. To complete the proof of (2.6) if suffices to show $Z_{TD} = Z^I$ and $Z_{BTC} = Z^I$.

To show $Z_{TD} = Z^I$, note that

(2.12) $P(TD) \cap P(CINB) = P(TDCNB)$

follows from $P(CINB) = P(CNB)$. To show $Z_{BTC} = Z^I$, note that

(2.13) $P(BTC) \cap P(DIN) = P(BTCDN)$

follows from $P(DIN) = P(DN)$.

Finally consider (2.7). $Z^{BI} = Z^{TBI}$ holds as (T) follows from (C) and (D). So it suffices to show $Z^{BI} = Z_{TC}^B$. Similar to the proof of (2.13), this follows from $P(DIN) = P(DN)$. $\square$

To complete the proof of the theorem, it suffices to exhibit instances showing that all the inequalities can be strict.

Lemma 2: There exist instances of the CPLP showing that each of the following inequalities can hold.

(2.14) $Z^{BI} < Z^I$

(2.15) $Z'_D < Z^B_C$

(2.16) $Z^T_C < Z^B_C$

(2.17) $Z^T_C < Z'_D$

(2.18) $Z_D < Z^T_C$

(2.19) $Z^B_C < Z^I$

(2.20) $Z_C < Z.$

Proof: First we give an example that satisfies (2.14) and (2.15).

$$(c_{ij}) = \begin{matrix} & & d = \\ 0 & 3 & 2 \\ 0 & 0 & 3 \end{matrix}$$

$$s = 3 \quad 5$$
$$f = 2 \quad 3$$

To compute Z^{BI}, we solve the corresponding linear program. Its optimum solution is $y_1 = 1$, $y_2 = 2/5$, $x_{11} = 1$, $x_{12} = 0$, $x_{21} = 1/3$ and $x_{22} = 2/3$, yielding the bound $Z^{BI} = 16/5$. Now consider Z^I. Its optimum solution is the same as for Z^{BI}, except that $y_2 = 2/3$. This yields $Z^I = 4$. Thus this example shows that (2.14) can occur.

Note that, for this instance, the constraint (T') used in Z'_D is $y_1 + y_2 \geq 1$ and therefore does not strengthen the relaxation Z^T_D. Since we have proved that $Z^T_D = Z^I$, we get $Z'_D = Z^T_D = Z^I = 4$.

Now we compute Z^B_C. Consider the Lagrangian relaxation

$$Z^B_{Cv} = \text{Min } \Sigma_i \Sigma_j c_{ij} x_{ij} + \Sigma_j f_j y_j + \Sigma_j v_j (\Sigma_i d_i x_{ij} - s_j y_j)$$

$$\Sigma_j x_{ij} = 1 \qquad \text{for all } i$$

$$\Sigma_j s_j y_j \geq \Sigma_i d_i$$

$$y_j \in \{0,1\} \qquad \text{for all } j.$$

If we set $v_1 = 2/3$, $v_2 = 0$, an optimum solution of the Lagrangian relaxation is $y_1 = y_2 = 1$, $x_{11} = 1$, $x_{12} = 0$, $x_{21} = 1/3$, $x_{22} = 2/3$. The corresponding value is $Z^B_{Cv} = 5$. Since we also have $Z = 5$, we deduce $Z^B_C = Z = 5$. Therefore the strict inequality (2.15) is satisfied by this instance of the CPLP.

To prove that (2.16) and (2.17) can hold, consider the previous example, where $s_2 = 5$ is changed into $s_2 = 4$. It is easy to check that this modification does not affect Z^B_C and Z^I. But, now, the constraint (T') used in Z'_D is $y_1 + y_2 \geq 2$ and therefore $Z'_D = Z^B_C = 5$. Next we compute $Z^T_C = \text{Max}_{v \geq 0} Z^T_{Cv}$, where

$$Z^T_{Cv} = \text{Min } \Sigma_i \Sigma_j c_{ij} x_{ij} + \Sigma_j f_j y_j + \Sigma_j v_j (\Sigma_i d_i x_{ij} - s_j y_j)$$

$$\Sigma_j x_{ij} = 1 \qquad \text{for all } i$$

$$0 \leq x_{ij} \leq y_j \qquad \text{for all } i, j$$

$$y_j \in \{0,1\} \qquad \text{for all } j.$$

This constraint set is totally unimodular as it is a simple plant location problem with two clients and two locations (see Cho, Padberg, Johnson and Rao[5] for example). Therefore $Z^T_C = Z^I = 4$. This proves that (2.16) and (2.17) can occur.

Next we give an example that satisfies (2.18).

$$(c_{ij}) = \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} \qquad \begin{matrix} d= \\ 1 \\ 1 \\ 1 \end{matrix}$$

$$s = 3 \quad 3 \quad 3$$
$$f = 1 \quad 1 \quad 1$$

The optimum value of the linear programming relaxation $Z^I = 3/2$ is obtained by taking the solution $y_j = 1/2$, $x_{ij} = 1/2$ if $i \neq j$ and $x_{ij} = 0$ if $i = j$.

Note that for this instance $Z_D^T = Z_D' = Z_D$ since there is no capacity constraint in effect. Since we have proved that $Z^T{}_D = Z^I$, we obtain $Z_D' = Z_D = Z^T{}_D = Z^I = 3/2$.

Similarly $Z^C = Z$ since the capacity constraints are automatically satisfied. This shows that $Z^C = Z_C = Z_C^T = Z = 2$.

Therefore this instance shows that possibility (2.18) can occur.

Finally we give an instance that satisfies (2.19) and (2.20).

$$(c_{ij}) = \begin{array}{ccc} 0 & 2 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 0 \end{array} \qquad \begin{array}{c} d= \\ 2 \\ 2 \\ 2 \end{array}$$

$$s = \quad 3 \quad 3 \quad 3$$

$$f = 3/2 \quad 3/2 \quad 3/2$$

To prove (2.19), note that $Z_C^B = 3$ since $y_1 = y_2 = y_3 = \frac{2}{3}$ and $x_{ij} = 1$ if $i=j$ and 0 if $i \neq j$ belongs to $P(CD) \cap P(INT)$. On the other hand, $Z^I = 4$ since $y_1 = y_2 = y_3 = \frac{2}{3}$ and $x_{11} = x_{22} = x_{33} = \frac{2}{3}$, $x_{13} = x_{21} = x_{32} = \frac{1}{3}$, $x_{12} = x_{23} = x_{31} = 0$ is an optimum solution of the linear programming relaxation. To prove (2.20), note that $Z = 9/2$ is obtained by setting $y_1 = y_2 = y_3 = 1$, $x_{ij} = 1$ if $i = j$ and 0 if $i \neq j$. For the solution of Z_C, consider $v_1 = v_2 = v_3 = 1/6$ in the Lagrangian relaxation Z_{Cv}. Then the same solution (x,y) is optimum. It yields the value $Z_{Cv} = 4$. Furthermore the subgradient at this point is negative in each component, so its projection on the nonnegative orthant is zero. This shows that $Z_C = 4$, proving (2.20).

This completes the proof of Lemma 2. $\qquad$ □

We close this section with results about variable splitting. It will be useful to generalize the notation introduced in Section 1. If the set of inequalities (S_1) consists of the constraint sets $(S_1^1),\ldots,(S_1^k)$ and (S_2) consists of $(S_2^1),\ldots,(S_2^\ell)$, then the bound defined by the split (S_1), (S_2) and X is denoted by $Z(S_1^1\ldots S_1^k, S_2^1\ldots S_2^\ell)$. As mentioned in the introduction, this value is obtained by minimizing cx + fy over

(2.21) $(x,y) \in P(S_1^1\ldots S_1^k\ X) \cap P(S_2^1\ldots S_2^\ell\ X)$.

Let us consider all the possible splits of the constraints (D), (C), (B), (T), (I) and (N) for CPLP. Our main result is that no new bound arises this way. Therefore variable splitting can be viewed as a different method for obtaining the bounds presented in Theorem 1. We make the following observations. If (I) is not in X, then only one of the two polyhedra in (2.21) is convexified by integality and therefore the corresponding bound is one of the Lagrangian dual bounds studied in Theorem 1. So we assume that (I) belongs to X. We also assume that (N) belongs to X as no benefit can be gained by doing otherwise. There are 25 possible splits of (D), (C), (B), (T), (I) and (N) with the property that (I) and (N) belong to X. If (S_1) or (S_2) is reduced to one of the sets (B), (T) or (B) U (T), then we do not get a true relaxation of CPLP, as one of the subproblems is CPLP itself. Therefore the true relaxations have the property that (D) belongs to (S_1) and (C) belongs to (S_2) or vice-versa. There are 9 such relaxations of CPLP. The bounds they produce equal Z_C or Z_D.

Theorem 2 The following equalities hold

(2.22) $Z(T,*) = Z(B,*) = Z(BT,*) = Z$

where * stands for any subset of the remaining constraints,

(2.23) $Z(C,D) = Z(C,DB) = Z(C,DT) = Z(C,DBT) = Z(D,TC) = Z(BD,TC) = Z_C$

(2.24) $Z(D,BC) = Z(D,TBC) = Z(TD,BC) = Z_D$.

Proof = (2.22) is immediate as one of the two subproblems is CPLP itself.

Next, we prove (2.23). Note that $Z(C,D)$, $Z(C,BD)$, $Z(C,DT)$ and $Z(C,DBT)$ are all at least as strong as Z_C, and $Z(C,D)$ is the strongest of the four. To prove $Z_C = Z(C,D)$ we show the polyhedral result $P(C) \cap P(DINTB) = P(CINTB) \cap P(DINTB)$. This follows from $P(CINTB) = P(C) \cap P(INTB)$ as we have already shown in (2.9), and from $P(DINTB) \subseteq P(INTB)$. To complete the proof of (2.23), it remains to show $Z(D,TC) = Z(BD,TC) = Z_C$. The feasible set for $Z(D,TC)$ is $P(DINB) \cap P(TCINB)$. Using (2.9), we have $P(CINTB) = P(B) \cap P(CINT)$ and therefore $Z(D,TC) = Z(BD,TC)$. Again, from (2.9) $P(DINB) \cap P(TCINB) = P(DINB) \cap P(INT) \cap P(C)$. In order to prove $Z(D,TC) = Z_C$, we must show $P(DINTB) = P(DINB) \cap P(INT)$. Let $(\bar{x},\bar{y})$ be an extreme point of $P(DINB) \cap P(INT)$. Note that $\{y: (x,y) \epsilon P(INT)$ for some $x\} \subseteq \{y: (x,y) \epsilon P(DINB)$ for some $x\} = \{y: \Sigma_j y_j \geq 1\}$ using the fact that $\Sigma_i d_i > 0$. Therefore $\bar{y}$ is an extreme point of the knapsack problem defined by (T), i.e. $\bar{y}$ is a 0,1 vector. As $(\bar{x},\bar{y}) \epsilon P(DINB)$, $\bar{x}$ is also a 0,1 vector. Thus $(\bar{x},\bar{y})$ is a 0,1 vector and $(\bar{x},\bar{y}) \epsilon P(DINTB)$.

Finally consider (2.24). $Z(D,BC)$ and $Z(D,TBC)$ are at least as strong as Z_D. To show that they are equal to it, it suffices to show that the feasible set for $Z(D,BC)$, namely $P(DINT) \cap P(BCINT)$, is identical to the feasible set for Z_D, namely $P(D) \cap P(BCINT)$. This follows from $P(DINT) = P(D) \cap P(INT)$ and $P(BCINT) \subseteq P(INT)$. Now consider $Z(TD,BC)$. It has feasible set $P(TDIN) \cap P(BCIN)$. The equalities $P(TDIN) = P(D) \cap P(INT)$ and $P(BCIN) = P(BC) \cap P(IN)$ imply $Z(TC,BC) = Z_D$. □

3. Computational Complexity

In Section 2, seven different bounds were obtained for CPLP, namely Z^{BI}, Z^I, Z_D', Z_D, Z_C^B, Z_C^T and Z_C. Here we consider the computational complexity of calculating these bounds. We also consider the complexity of computing the optimum value Z of CPLP itself.

Theorem 3

(3.1) Z^{BI}, Z^I and Z_D' can be computed in polynomial time.

(3.2) Z_D and Z_C^B can be computed in pseudo-polynomial time.

(3.3) Computing Z_C^T, Z_C or Z is strongly NP-hard.

Proof = Linear programs can be solved in polynomial time, so (3.1) holds for Z^{BI} and Z^I. Z_D' is obtained by minimizing $cx + fy$ over $P(D) \cap P(CBINT') = P(DCB) \cap P(INT')$. The constraint set (N), (T') is totally unimodular. So computing Z_D' reduces to the solution of a linear program. This proves (3.1).

Computing Z_D or Z_C^B is NP-hard since, when $c=0$, the problem is nothing but a knapsack problem. Next we show that Z_D can be computed in pseudo-polynomial time. To get Z_D, we optimize $cx + fy$ over $P(DCB) \cap P(INT)$. If

follows from a result of Gröschel, Lovasz and Schrijver [15] that optimizing over $P(DCB) \cap P(INT)$ requires at most pseudo-polynomial time since optimizing over $P(DCB)$ is polynomial and optimizing over $P(INT)$ is pseudo-polynomial. To get z_C^B, we optimize $cx + fy$ over $P(DC) \cap P(INT)$. so the same argument can be used. This proves (3.2).

Finally, (3.3) follows from the fact that SPLP is strongly NP-hard and polynomially reduces to CPLP by setting the capacities s_j large enough. □

Although CPLP is NP-hard, there are some special cases that can be solved in polynomial time. In the next theorem, we show that CPLP can be solved in polynomial time when the capacities and the demands are taken from the set $\{1,2\}$ and it is NP-hard when the capacities are taken from the set $\{1,\ldots,p\}$, $p \geq 3$ even if all the demands are equal to 1. We also consider the complexity of computing Z_D, z_C^B, z_C^T and Z_C under the last assumption.

Theorem 4

(3.4) CPLP is strongly NP-hard when the capacities are taken from the set $\{1,\ldots,p\}$ for any fixed $p \geq 3$, even if all the demand are equal to 1. Under this assumption, Z_D or z_C^B can be computed in polynomial time but computing z_C^T or Z_C is strongly NP-hard.

(3.5) CPLP can be solved in polynomial time when the capacities and demands are taken from the set $\{1,2\}$.

Proof: We will prove that CPLP is strongly NP-hard for $p \geq 3$ by reducing the 3-dimensional matching problem (3DM) to CPLP.

3DM is defined by three distinct sets I, J, K, each of cardinality n, and a family F of triplets (i,j,k) where $i \in I$, $j \in J$ and $k \in K$. The question

is whether there exists $S \subseteq F$ of cardinality n such that the 3n elements of I, J, K each appear exactly once in the triplets of S. This problem is NP-complete (Garey and Johnson [12]). We construct a CPLP as follows. There are 3n clients corresponding to the elements of I, J, K and n^3 facilities corresponding to all the triplets (i, j, k) with $i \in I$, $j \in J$, $k \in K$. The transportation costs from a triplet (i, j, k) $\in$ F to i, j and k are equal to 0. All other transportation costs are equal to 1. The capacities are equal to 3, the demands to 1 and the fixed costs to $\varepsilon < \frac{1}{n}$. 3DM has a solution if and only if CPLP has a solution with value n ε.

To see that Z_D or Z_C^B can be computed in polynomial time, note that $0 \le \Sigma_i d_i \le \Sigma_j s_j \le pn$, where, here, n denotes the number of facilities. Since p is fixed, a dynamic programming algorithm for solving the knapsack problem defined by (T) is polynomial. The result now follows from Gröschel, Lovasz and Schrijver [15].

To see that Z_C^T and Z_C are NP-hard, note that the transformation of 3DM described above is valid even if we relax the capacity constraints (C) and (T) completely. 3DM has a solution if and only if Z^{TC} has the value nε. This completes the proof of (3.4).

Now we prove (3.5). From an instance of CPLP where the demands and capacities are taken from {1,2}, we construct a graph as follows. For each customer i with demand d_i we construct d_i nodes, for each facility j with capacity s_j we construct s_j nodes and for each facility j with capacity 1 we construct an additional node. The corresponding node sets are denoted by U, V and W respectively. The graph has an edge with cost c_{ij} joining each node of U associated with customer i to each node of V associated with facility j. For each facility j such that $s_j = 2$, the two nodes of V associated with j are joined by an edge with cost $-f_j$. Finally, each node of V associated with a

facility with $s_j = 1$ is joined to the corresponding node of W by an edge of cost $- f_j$. Now CPLP is equivalent to solving a minimum cost matching problem in this graph, where each node of U is required to be exactly matched and each node of V U W must meet at most one edge of the matching. (The closed facilities correspond to the nodes of V that are matched to a node of W or to another node of V). (3.5) follows from the fact that the matching problem can be solved in polynomial time (Edmonds[9]).   □

It is interesting to note that, although Z can be computed in polynomial time under the assumption made in (3.5), it is not clear whether Z_C and Z_C^T can. In fact, there are values of the Lagrange multipliers for which the Lagrangian relaxations Z_{Cv} and Z_{Cv}^T are NP-hard to solve. (For example, the reduction of 3DM used for proving (3.4) is still valid when $v = 0$.)

4. Computational Results

Our computational study compares the relaxations Z^I, Z_D, Z_C^T and Z_C to the optimum value Z of CPLP. We also compare a dual-based heuristic to greedy and interchange heuristics. One objective of the study is to test the tightness of the bounds as a function of parameters of the problem. In particular, we consider the influence of tight capacity constraints.

Z_D is computed using a subgradient algorithm stopped after a maximum of 200 iterations. The knapsack subproblems are solved using the algorithm of Fayard and Plateau[11]. To compute Z^I, we solve the Lagrangian dual giving Z_{DC}^T by a subgradient algorithm (recall that $Z^I = Z_{DC}^T$). To obtain Z_C^T, Z_C and Z, we implement the cross decomposition of Van Roy [30]. The optimum value Z is computed using a branch-and-bound algorithm based on the bound Z_C. The bound Z_C itself is calculated by solving iteratively a sequence

of problems Z_{Cu}. For each value of u, Z_{Cu} is obtained using a branch-and-bound algorithm based on Z_{Cu}^T. Finally, the simple plant location problem that yields Z_{Cu}^T is solved using the dual ascent procedure of Erlenkotter [10] which also involves branch and bound. Clearly, due to the triply nested branch-and-bound algorithms, cross decomposition is most effective when the duality gap $Z - Z_C^T$ is small. Dual heuristics are needed to accelerate the search for the optimal Lagrange multipliers. Lacking this, it is not obvious how to generate good dual information from the primal subproblems. For the harder problem instances, the cross decomposition method often reduces to a rather inefficient column generation method, as corroborated by Van Roy.

On the heuristic side, we implemented DROP-HI and VSM - a greedy heuristic starting from all facilities open and an interchange heuristic, respectively - as described in Jacobsen[20]. The implementation suggested by Khumawala [21] gives similar, although slightly inferior, results. The computational results provided below follow Jacobsen [20]. We also consider a dual-based heuristic, denoted by (H). At each step of the subgradient algorithm for computing Z_D, a solution (x_D, y_D) of the Lagrangian subproblem is found. If it is feasible for the original problem, it is an optimum solution. Otherwise, consider y_D as the set of open facilities and solve a transportation problem to find a feasible solution $(\bar{x}_D, y_D)$. The best such solution found during the subgradient iterations is taken as the solution of heuristic (H). We have observed that, typically, only few different vectors y_D are generated during the algorithm and so only few transportation problems have to be solved. We use the code of Srinivasan and Thompson [29] for solving the transportation problems.

To validate our algorithms, we solve the classical test problems of Kuehn and Hamburger [23] and are able to confirm the computational results already

reported in the literature. For example our evaluation of $\frac{Z-Z_C}{Z}$ coincides with ϵ^{OPT} in table II of Van Roy [30] in all but one of the instances. (In problem 24, we found .18% instead of .21% but Van Roy truncated his branch-and-bound after 2000 nodes whereas we did not.) Van Roy does not report Z_C^T. We found that $Z_C^T = Z_C$ for all the problems except problems 11 and 12, where $\frac{Z-Z_C^T}{Z}$ = .074% and .277% respectively. Regarding Z_D, our results are better than expected. The reason is that, although Nauss [25] recommends to compute the bound Z_D in his paper, he reports computational results that are not based on Z_D. More specifically, Nauss relaxed the integrality requirements in the knapsack subproblems needed to compute Z_D (see first sentence of second paragraph in section COMPUTATIONAL RESULTS of [25]). The reason for doing this is unclear, since the knapsack subproblems are very easy to solve when all the s_j's are equal. As a result, Nauss computed Z^I instead of Z_D. Indeed, in Table I of [25], the gaps closed by LGR1 and LGR2 after subgradient optimization are very similar. (By Theorem 1, they should eventually be identical, after sufficiently many steps of the subgradient algorithm.) By solving true knapsack subproblems, we find that Z_D often greatly improves on Z^I. For example, $Z_D = Z$ for problems 6, 8 and 9 as numbered by Nauss. Going back to the numbering of Van Roy, we find that Z_D = Z for problems 1-6. For the more tightly capacitated problems 7-12, we find $\frac{Z-Z_D}{Z}$ = .06%, .20%, .64%, .95%, .66% and .36% respectively. For these 12 instances, the heuristic (H) always finds the optimum solution. The heuristics DROP-HI and VSM perform reasonably well (see Jacobsen [20]).

To complement these results, we run the algorithms on a different and larger set of test problems.

We generate 5 sets of 30 problems. In each set, there are 5 problems of each of the following sizes - 25x8, 25x16, 25x25, 50x16, 50x33, 50x50 where

the first number is the number of clients and the second is the number of facilities. The transportation costs are computed by generating points representing the clients and facilities uniformly in a unit square. The Euclidean distance between them is multiplied by 1C to define unit transportation costs. The demands are generated from $U[5,35]$, where $U[a,b]$ denotes the uniform distribution in the interval $[a,b)$. The capacities s_j are generated from $U[10,160]$ and the fixed costs by the formula $f_j = U[0,90] + U[100,110]$ $\sqrt{s_j}$ to reflect economies of scale. From the above data, 5 different sets of problems are generated by scaling the capacities using the ratios $\frac{\Sigma_j s_j}{\Sigma_i d_i} = 1.5, 2, 3, 5, 10$. Finally f_i is doubled for the first 2 sets. Although the above choices are somewhat arbitrary, our objective is to generate a reasonably difficult set of test problems. The actual data are available from the authors.

We report our computational experience with these 5 sets of problems in Tables I-V. In each case we give the relative error of the bound to the optimum value. For a relaxation such as Z_C we report $\frac{Z-Z_C}{Z}$, and for a heuristic with value $Z(H)$ we report $\frac{Z(H)-Z}{Z}$.

Table I $\quad \Sigma_j s_j = 1.5 \ \Sigma_i d_i$

Relative Error %	Relaxations				Heuristics		
	z_C	z_D	z_C^T	z^I	H	VSM	DROP-HI
25 x 8	0	0	5.70	5.88	0	0	17.27
	0	0.01	6.38	6.85	0	5.73	5.73
	0	0.02	5.85	6.67	0	6.22	11.14
	0	0	1.30	5.10	0	13.13	13.13
	0	0.02	5.48	5.71	0	8.82	8.82
25 x 16	0.37	0.37	3.05	3.39	0	15.69	15.69
	0	0	3.12	14.32	0	6.23	20.80
	0	0	2.14	15.07	0	13.30	13.30
	0.12	0.12	1.63	2.49	0	10.92	15.20
	0.01	0.07	3.09	20.87	0	0.29	5.36
25 x 25	0	0.01	1.41	46.88	0	5.95	30.01
	0	0.01	0.54	2.05	0	10.02	20.96
	0.04	0.09	1.65	2.30	0	9.18	26.96
	0	0	1.93	15.28	0	15.01	38.25
	0.07	0.09	1.55	29.27	0	12.57	26.36
50 x 16	0	0.01	2.18	4.30	0	15.30	17.97
	0	0	0.42	1.23	0	17.75	23.71
	0	0.01	2.27	2.45	0	4.08	24.42
	0.26	0.29	2.07	2.10	0	2.77	16.78
	0	0	0.91	9.27	0	10.48	22.86
50 x 33	0.10	0.14	1.48	3.49	0	25.52	26.24
	0.05	0.10	1.17	17.33	0	19.35	24.10
	0	0.16	0.18	0.59	0	18.58	28.02
	0.12	0.14	1.03	3.01	0	5.77	26.10
	0.05	0.09	1.61	2.89	0	11.72	29.98
50 x 50	0	0.06	0.25	1.45	0	*	27.63
	0	0.07	0.09	19.62	0	*	26.08
	0	0.02	0.38	32.96	0	*	24.44
	0.01	0.02	0.19	0.41	0	*	28.88
	0	0.10	0	11.80	0	*	27.58
Average %	0.04	0.07	1.97	9.83	0	10.57	21.46

* time limit of 120 seconds on DEC 20 exceeded.

Table II $\quad \Sigma_j s_j = 2 \Sigma_i d_i$

Relative Error %	Relaxations				Heuristics		
	Z_C	Z_D	Z_C^T	Z^I	H	VSM	DROP-HI
25 x 8	0	0	5.94	6.22	0	0	24.05
	0.75	0.79	10.31	10.32	0	0.82	8.40
	0	0.05	7.48	7.80	0	4.27	14.40
	0	0.02	2.23	2.25	0	0	35.71
	0	0	5.24	5.26	0	19.54	21.84
25 x 16	0.19	0.19	4.91	4.97	0	10.36	41.98
	0.04	0.09	1.43	1.47	0	13.00	22.36
	0	0	2.09	2.27	0	8.75	38.64
	0.15	0.15	4.36	4.36	0	14.30	20.91
	0	0	2.74	3.34	0	18.59	44.04
25 x 25	0.05	0.12	2.28	2.28	0	4.82	34.68
	0.18	0.23	1.42	1.57	0	9.56	29.89
	0	0.04	0.78	0.84	0	20.25	40.33
	0	0.03	1.76	2.38	0	13.30	37.25
	0.11	0.12	3.04	3.07	0	6.76	41.36
50 x 16	0	0.19	3.66	3.89	0	13.19	27.92
	0.01	0.17	2.55	2.57	0	5.93	23.11
	0	0	0.46	0.49	0	11.05	25.08
	0.42	0.43	2.45	11.85	0	14.12	14.12
	0	0	0	0	0	7.61	41.27
50 x 33	0.04	0.04	1.50	1.50	0	10.84	46.67
	0.09	0.09	1.15	1.19	0	11.81	32.34
	0.08	0.10	1.52	1.86	0	19.98	47.39
	0.01	0.02	0.67	0.81	0	14.38	28.48
	0.09	0.09	1.34	1.34	0	12.82	39.08
50 x 50	0	0	0.23	0.34	0	*	36.27
	0.21	0.27	1.01	1.25	0.01	*	41.60
	0	0.02	0.14	1.43	0	*	45.38
	0	0.03	0.13	0.36	0	*	39.02
	0	0.04	0.02	0.13	0	*	42.78
Average %	0.08	0.11	2.43	2.91	0	10.64	33.21

Table III $\quad \Sigma_j s_j = 3\, \Sigma_i d_i$

Relative Error %	Relaxations				Heuristics		
	Z_C	Z_D	Z_C^T	Z^I	H	VSM	DROP-HI
25 x 8	0	0	7.06	8.94	0	0	12.85
	0	0	1.47	1.71	0	12.10	22.81
	0	0	5.69	5.91	0	0	3.92
	0	0	4.98	5.07	0	8.97	16.11
	0	0	4.13	4.63	0	10.38	30.23
25 x 16	0	0.04	6.00	6.10	0	2.90	39.29
	0	0	1.45	1.47	0	9.65	39.18
	0	1.48	4.38	8.44	0	3.13	39.35
	0	0.25	5.44	5.54	0	0	60.09
	0	0.09	3.65	3.65	0	0	39.47
25 x 25	0.11	0.11	0.39	0.41	0	16.97	62.02
	0.20	0.33	4.60	4.63	0.09	0.69	53.54
	0	0.06	0.45	0.45	0	1.56	53.93
	0.92	1.19	2.06	2.09	0	17.82	52.53
	0	0.44	2.18	2.26	0	0	61.60
50 x 16	0.19	0.82	4.10	4.71	0	6.90	33.88
	0.13	1.55	5.55	5.66	0	0.63	15.29
	0	0	1.34	1.41	0	0	26.03
	0	0	1.95	2.40	0	3.35	26.83
	0	0	2.08	2.08	0	0	41.84
50 x 33	0.03	0.46	2.40	2.51	0	11.40	53.72
	0.56	0.70	2.90	2.93	0	12.37	54.61
	0.76	0.92	1.53	1.59	0	8.38	47.45
	0.27	0.30	0.76	0.79	0	13.02	40.10
	0.41	0.75	2.06	2.06	0	2.35	53.66
50 x 50	0.17	0.26	1.43	1.43	0	4.20	64.06
	0.52	0.55	1.14	1.16	0	*	62.71
	0.29	0.51	0.88	0.88	0	11.71	44.40
	0.30	0.35	0.52	0.56	0	*	49.08
	0.24	0.39	0.71	0.73	0	*	44.47
Average %	0.16	0.38	2.78	3.07	0	5.87	41.50

Table IV $\Sigma_j s_j = 5 \Sigma_i d_i$

Relative Error %	Relaxations				Heuristics		
	Z_C	Z_D	Z_C^T	Z^I	H	VSM	DROP-HI
25 x 8	0	0	11.61	11.75	0	0	3.92
	0	0	5.95	5.95	0	3.98	10.64
	0	0.31	5.39	5.65	0	0	18.88
	0	2.28	11.40	11.66	0	0	11.90
	0	0	0	0	0	0	23.92
25 x 16	0	0	1.57	2.07	0	0	59.50
	0	0	0.22	0.82	0	1.61	49.81
	0	0	3.72	4.06	0	4.55	37.36
	0	0	1.76	2.01	0	3.18	25.06
	0.12	0.12	1.54	2.34	0	0	40.53
25 x 25	0	1.07	3.76	3.89	0	3.33	52.94
	0.19	0.74	2.68	12.10	0	13.31	32.80
	0.27	2.57	5.90	5.90	0.07	7.11	59.22
	0.30	0.53	2.43	2.90	0	0	64.85
	0	0.22	2.11	2.20	0	2.22	34.07
50 x 16	0	0	0.51	1.40	0	0	30.36
	0	0.70	3.27	3.68	0	2.37	28.91
	0.38	0.38	1.87	3.92	0	0	31.11
	0.65	2.10	4.73	4.90	0	0	22.01
	0	0.85	4.13	4.31	0	1.76	25.55
50 x 33	0	0.53	1.79	1.87	0	0	50.15
	0	0.01	0.81	0.91	0	5.46	68.73
	0	0.29	1.25	1.27	0	0	52.80
	0.07	0.15	0.92	1.17	0	1.93	69.54
	0.34	1.16	2.12	2.20	0	3.78	45.31
50 x 50	0.35	0.51	1.27	1.30	1.13	8.56	64.41
	0.41	1.34	2.80	2.84	1.24	0.92	79.34
	0.22	0.82	2.45	2.45	0.01	17.38	58.11
	0.01	0.43	0.89	0.89	0.42	0.26	67.37
	0.48	0.68	2.48	2.48	2.02	1.37	52.14
Average %	0.13	0.59	3.04	3.63	0.16	2.77	42.37

Table V $\quad \Sigma_j s_j = 10 \Sigma_i d_j$

Relative Error %	Relaxations				Heuristics		
	Z_C	Z_D	Z_C^T	Z^I	H	VSM	DROP-HI
25 x 8	0	0	6.06	6.10	0	0	17.68
	0	0	0	0.31	0	0	24.90
	0	0	0	0	0	0	22.09
	0	0	6.04	7.31	0	2.10	3.32
	0	0.21	3.51	3.69	0	0	1.91
25 x 16	0	0	6.20	6.35	0	0	22.37
	0	0	8.24	9.82	0	0	35.20
	0	0	9.08	9.39	0	0	51.02
	0	0	0	0.61	0	0	58.71
	0	0	0	0	0	0	34.97
25 x 25	0	0	1.12	1.21	0	0	48.54
	0	0.43	10.41	11.32	0	5.39	64.99
	0.05	0.08	3.74	3.74	0	3.05	41.00
	0	0	5.35	5.90	0	1.55	51.62
	0	0	5.36	8.78	0	8.41	80.28
50 x 16	0	0.13	4.13	4.64	0	0	18.02
	0	1.24	6.31	7.77	0	0	7.52
	0	2.02	4.46	6.26	0	0	19.63
	0	0	0	0	0	0	21.89
	0	0	1.80	2.37	0	0	27.11
50 x 33	0	0	2.23	14.10	0	10.50	62.96
	0	0	0.27	0.29	0	0	50.59
	0	0.13	2.33	5.36	0	0	37.67
	0	0.72	1.77	1.88	0	7.98	48.65
	0	0	0.35	0.49	0	0.94	58.59
50 x 50	0.01	0.51	2.31	10.94	0	1.07	77.06
	0	0	0.81	0.94	0	10.22	81.07
	0.04	0.83	3.50	4.32	0	0	76.16
	0.02	0.30	1.56	2.01	.10	0	75.74
	0.05	0.90	3.85	4.59	.22	5.62	63.11
Average %	0.01	0.25	3.36	4.68	0.01	1.88	42.81

Some remarks can be made about Tables I-V. The relaxations Z_C and Z_D are both significantly stronger than Z_C^T and Z^I for this set of test problems. Z_C is about twice as strong as Z_D for the tightly capacitated problems and this advantage increases for the less capacitated ones. The quality of both bounds deteriorates in the intermediate range, i.e. $\Sigma_j s_j / \Sigma_i d_i = 3$ and 5, but Z_C

never exceeds 1% and Z_D never exceeds 3%. Whereas for the Kuehn and Hamburger test problems Z_C^T is frequently equal to Z_C, here Z_C^T is typically much weaker. For the tightly capacitated problems, the negative effect of removing the knapsack constraint (T) is more pronounced when there are few facilities (e.g. 8 or 16). Our subgradient algorithm for computing Z^I had difficulties converging for the tightly capacitated problems of Table I. So the large gaps appearing there should be interpreted with caution. DROP-HI and VSM are the two heuristics recommended by Jacobsen, based on tests with the Kuehn and Hamburger problems. Their performance is very different when the facilities do not have identical capacities and fixed costs, as pointed out by Domschke and Drexl [8]. Our own computational experience serves to reiterate this warning. DROP-HI tends to close the large facilities first, clearly not a good strategy. VSM performed reasonably well on the less capacitated problems. However, the clear winner is heuristic (H), failing to find the optimum solution in only 10 out of 150 cases. This is particularly remarkable considering that the time needed to compute Z_D and to find the heuristic solution (H) only grows moderately with problem size. By contrast, VSM exceeded the time limit of 120 seconds on a DEC 20 in thirteen instances. Computing times are provided in Table VI. More extensive computational results are reported in Sridharan [28].

Table VI Computing Times (Average of 25 problems).

Problem Size	Seconds on IBM 3081		Seconds on DEC 20			
	Z_C	Z_C^T	Z^I	Z_D and Z_H	VSM	DROP-HI
25 X 8	0.9	1.0	2.1	1.7	0.6	0.1
25 X 16	14	12	5.7	3.9	2.8	0.2
25 X 25	140	69	11	9.0	9.6	0.5
50 X 16	34	23	8.7	9.6	5.7	0.5
50 X 33	1900	570	28	23	41	1.7
50 X 50	6800	5200	62	37	>92*	3.7

*13 instances exceeded the time limit of 120 seconds on a DEC 20.

5. Probabilistic Results

In this section we assume that:

(i)   n clients are distributed independently and uniformly in the unit square,

(ii)   every client location is also a potential site for locating a facility,

(iii) c_{ij} is proportional to the Euclidean distance between points i and j,

(iv)   all the clients have the same demand, independent of n,

(v)   all the facilities have the same fixed cost $f = f(n)$ and the same capacity $s = s(n)$.

(vi) $s/\log n \to \infty$ as $n \to \infty$,

(vii) $n^{\varepsilon - 1/2} \leq f \leq n^{1-\varepsilon}$ for some fixed $\varepsilon > 0$.

Although these assumptions are restrictive, we believe that they represent an interesting starting point for a probabilistic comparison of heuristics and relaxations for the CPLP. In fact, the results we are about to present still hold under weaker assumptions. In (i), the shape of the region can be almost anything (the neighborhood of the boundary must be of measure 0) and the distribution can be nonuniform as long as mass is not concentrated. In (ii), the potential sites need not coincide with the clients as long as they have a similar distribution. In (iii), the Euclidean distance can be computed after a linear transformation of the region. In (iv), the demands can be different for each client as long as they are bounded independently of n. In (v), the fixed values f and s can each be replaced by a small range of values. The assumption in (vii) can be slightly relaxed to $\omega/\sqrt{n} \leq f \leq n/\omega$ where $\omega \to \infty$ as $n \to \infty$. Our goal in this section is only to give a flavor of typical results and assumptions (i)-(vii) serve this purpose.

Without loss of generality, we take the constant of proportionality in (iii) to be 1 and the demand in (iv) to be 1. Then, given f and s considered as functions of n, the value $Z = Z(n)$ is a random variable for every $n \geq 1$. Similarly, relaxations or heuristics yield random variables $Z_R(n)$ or $Z_H(n)$ for every $n \geq 1$. In this section we study the probabilistic convergence of such sequences of random variables as $n \to \infty$.

First, we introduce some notation. Let $W(n)$ be a sequence of random variables for $n \geq 1$. We say that $W(n) \to w$ _almost surely_ if and only if

(5.1) $\Pr[\limsup W(n) = w = \liminf W(n)] = 1.$

If we only have $\Pr[\limsup W(n) \leq w] = 1,$ we say that $W(n) \leq w$ almost surely. A sufficient condition for (5.1) is (Borel-Cantelli lemma)

(5.2) $\sum_{n=1}^{\infty} \Pr[|W(n)-w| > \varepsilon] < \infty,$ for every fixed $\varepsilon > 0$.

Let $V(n)$ and $W(n)$ be two sequences of random variables for $n \geq 1$. We write $W(n) \sim V(n)$ almost surely if and only if $W(n)/V(n) \to 1$ almost surely. Frequently, we simply write $W \sim V$ almost surely.

In this section, we show that the relaxations of the CPLP introduced earlier fall into 3 classes according to the probabilistic convergence of $(Z-Z_R)/Z$. Let $\alpha = .32887076\ldots.$

Theorem 5

Assume (i) - (vii). Then

(a) $Z^T_C \sim Z$ almost surely,

(b) $Z_D \sim Z^I$ almost surely and

$Z^I/Z \geq .99810744$ almost surely,

(c) $Z^{BI} = fn/s$ and

$Z^{BI}/Z \leq .33333334$ almost surely for $f \leq (\alpha s)^{3/2}/\sqrt{n}$.

For $f > (\alpha s)^{3/2}/\sqrt{n}$, the ratio Z^{BI}/Z improves above 33% but the bound Z^{BI} is still significantly weaker than the ones considered in (a) and (b). The 3 parts of the theorem are proved in Lemmas 6, 7 and 8 respectively. Our analysis uses the following results.

Lemma 3: (Hoeffding [19]) If $Y_1, \ldots, Y_n$ are independent random variables and $0 \le Y_i \le 1$ for $i = 1, \ldots, n$, then, for $0 \le \varepsilon \le 1$,

$$\Pr[\sum_{i=1}^{n} Y_i \ge (1+\varepsilon)n] \le \varepsilon^{-\varepsilon^2 n\mu/3}$$

where μ is the expected value of $(\sum_{i=1}^{n} Y_i)/n$.

Papadimitriou [26] introduced the following K-honeycomb heuristic.

Partition the plane into a regular hexagonal tiling, where each hexagon has the same area $1/K$. For each hexagon H_q that intersects the set of clients, let P_q be the client closest to its center. Choose this set $\{P_q\}$ as the set of open facilities. We denote by Z_K the value of this heuristic.

Papadimitriou [26] proved that Z_K is near-optimal for the K-median problem. A similar result holds for the SPLP for an appropriate choice of K, as shown in Ahn, Cooper, Cornuejols and Frieze [1].

Lemma 4: Assume (i) - (vii) and $s = \infty$. Then

(a) $Z_K \sim Z$ almost surely for $K \sim \alpha(n/f)^{2/3}$,

(b) $Z \sim 3\alpha(fn^2)^{1/3}$ almost surely.

To approximate the linear programming relaxation of SPLP, we define the following feasible fractional solution for each $K=1,\ldots,n$. For $j=1,\ldots,n$, set $y_j = K/n$. For $i=1,\ldots,n$, define j^* to be a facility such that $c_{ij} < c_{ij}^*$ for at most n/K values of j and $c_{ij} \leq c_{ij}^*$ for at least $n/K + 1$ values of j; take $x_{ij} = K/n$ for n/K facilities $j \neq j^*$ such that $c_{ij} \leq c_{ij}^*$, take $x_{ij}^* = 1 - n/K \ \ K/n$ and $x_{ij} = 0$ for every other value of j. We call this solution the K/n-uniform fractional solution and denote by $Z_F = \Sigma_i \Sigma_j c_{ij} x_{ij} + Kf$ its value.

Lemma 5: (Ahn, Cooper, Cornuejols and Frieze [1]) Assume (i) - (vii) and $s=\infty$. Then

(a) $Z_F \sim Z^I$ almost surely for $K \sim \beta(n/f)^{2/3}$,

(b) $Z^I \sim 3\beta(fn^2)^{1/3}$ almost surely,

where $\beta = .32824836 \ldots = 1/(9\pi)^{1/3}$.

Lemma 6: $Z^T_C \sim Z$ almost surely.

Proof: First, we find a lower bound on Z^T_C.

$$Z^T_C = \underset{v_j \geq 0}{\text{Max}} \ \text{Min} \ \Sigma_i \Sigma_j (c_{ij} + v_j) x_{ij} + \Sigma_j (f - sv_j) y_j$$
$$\Sigma_i x_{ij} = 1 \quad \text{for all } j$$
$$0 \leq x_{ij} \leq y_j \quad \text{for al i, j}$$
$$y_j \in \{0,1\} \quad \text{for all j.}$$

Setting $v_j = v$ for all j, we get, for every $v \geq 0$,

(5.3) $Z_C^T \geq nv + Z(v)$ where

$$Z(v) = \text{Min } \Sigma_i \Sigma_j c_{ij} x_{ij} + \Sigma_j (f-sv) y_j$$
$$\Sigma_i x_{ij} = 1 \qquad \text{for all } j$$
$$0 \leq x_{ij} \leq y_j \qquad \text{for all } i, j$$
$$y_j \in \{0,1\} \qquad \text{for all } j.$$

This is an SPLP. Let $\epsilon \to 0$ as $n \to \infty$. The actual value of ϵ will be given later. By Lemma 4(a), the K-honeycomb heuristic satisfies

(5.4) $Z_K \sim Z(v)$ almost surely for K and v defined as follows :

$K = n(1+\epsilon)/s = \alpha(n/(f-sv))^{2/3}$ when $\alpha(n/f)^{2/3} \leq n(1+\epsilon)/s$

$K = \alpha(n/f)^{2/3}$ and $v = 0$ when $\alpha(n/f)^{2/3} > n(1+\epsilon)/s$.

We claim that the K-honeycomb heuristic is almost surely feasible to CPLP for this choice of K and v. To prove this, it suffices to show that there are almost surely at most s clients in each hexagon. The number N of clients in an hexagon of area 1/K has the binomial distribution B(n; 1/K). By Lemma 3,

$$\Pr[N \geq (1+\epsilon)n/K] \leq e^{-\epsilon^2 n/3K}.$$

Let E be the event that at least one hexagon of the tiling contains more than s clients. Since $K \rightarrow \infty$, the number of hexagons of the tiling that intersect the unit square is $(1+o(1))K$. Thus

$$\Pr[E] \leq (1+o(1))Ke^{-\epsilon^2 n/3K}.$$

Let us choose ϵ so that $\epsilon^2 n/3K = 3\log n$. It is easy to check that $\epsilon \rightarrow 0$ as $n \rightarrow \infty$. So

$$\Pr[E] \leq (1 + o(1))K/n^3 \leq 1/n^2.$$

This shows that the K-honeycomb heuristic is almost surely feasible for CPLP for the above choice of K and v. This implies

(5.5) $Z \leq Z_K + Ksv$ almost surely.

The inequalities (5.3), (5.4), (5.5) imply

$$Z_C^T \geq nv + Z_K (1-o(1)) \geq Z(1-o(1) + nv - Ksv(1-o(1)) \text{ almost surely.}$$

Since $n = Ks(1-o(1))$ and $nv \leq Z$, we get

$$Z_C^T \geq Z (1-o(1)) \text{ almost surely.}$$

Since Z_C^T is a relaxation of CPLP, $Z_C^T \leq Z$. So

$$Z_C^T \sim Z \text{ almost surely.} \qquad \square$$

Lemma 7:

$Z^I/Z \rightarrow .99810744 \ldots$ almost surely when $f \le s^{3/2}/3\sqrt{\pi}n$

$Z^I/Z \ge .99810744$ almost surely when $f > s^{3/2}/3\sqrt{\pi}n$

$Z_D \sim Z^I$ almost surely.

Proof:

$$Z^I = \min \Sigma\Sigma c_{ij}x_{ij} + \Sigma f y_j$$
$$\Sigma_j x_{ij} = 1$$
$$\Sigma_i x_{ij} \le s y_j$$
$$0 \le x_{ij} \le y_j \le 1.$$

Consider the K/n-uniform fractional solution of Lemma 5, for $K = \beta(n/f)^{3/2}(1+\epsilon)$ where $\epsilon \rightarrow 0$.

Assuming $f \le s^{3/2}/3\sqrt{n}$, we get $n/K \le s(1+\epsilon)$.

So this solution satisfies the capacity constraints and therefore is feasible to Z^I, i.e. $Z^I \le Z_F$.

By Lemma 5, $Z_F \sim Z^{CI} \sim 3\beta(fn^2)^{1/3}$ almost surely.

Since $Z^{CI} \le Z^I$, we get

$Z^I \sim 3\beta(fn^2)^{1/3}$ almost surely.

The inequality $f \le s^{3/2}/3\sqrt{\pi}n$ implies $\alpha(n/f)^{3/2} \ge n(1+\epsilon)/s$. We have shown in Lemma 6 that, under this condition, the K-honeycomb heuristic is almost surely feasible to CPLP for $K = \alpha(n/f)^{2/3}$. By Lemma 4, this implies

$Z \sim 3\alpha(fn^2)^{1/3}$ almost surely.

Thus, when $f \leq s^{3/2}/3\sqrt{\pi}n$, $Z^I/Z \sim \beta/\alpha$ almost surely.

Since $\beta/\alpha = .99810744 \ldots$, we have shown the first part of Lemma 7.

When $f > s^{3/2}/3\sqrt{\pi}n$, note that we can write

$$Z^I = \max_{v \geq 0} \min \Sigma\Sigma c_{ij}x_{ij} + \Sigma(f-sv)y_j + nv$$
$$\Sigma_j x_{ij} = 1$$
$$\Sigma_i x_{ij} \leq sy_j$$
$$0 \leq x_{ij} \leq y_j \leq 1.$$

Now consider the K/n-uniform fractional solution for $K = (n/(f-sv))^{2/3}(1+\epsilon)$ where $\epsilon \to 0$. By Lemma 5, its value satisfies:

$$Z_F \sim 3\beta[(f-sv)n^2]^{1/3} \text{ almost surely.}$$

Therefore

$$Z^I \geq \max_{v \geq 0} 3\beta[(f-sv)n^2]^{1/3} + nv \text{ almost surely.}$$

This function of v has a unique maximum, obtained when:

$$(5.6) \quad n/s = \beta(n/(f-sv))^{2/3}.$$

This implies $n/K = s/(1+\epsilon) \leq s$. So the above K/n-uniform fractional solution is feasible for Z^I. This shows that

$$Z^I \sim 3\beta[(f-sv)n^2]^{1/3} + nv \quad \text{almost surely}$$

where v is defined by equation (5.6).

Similarly

$$Z = \max_{v \geq 0} \min \Sigma\Sigma c_{ij}x_{ij} + \Sigma(f-sv)y_j + nv$$
$$\Sigma_j x_{ij} = 1$$
$$\Sigma_i x_{ij} \leq sy_j$$
$$0 \leq x_{ij} \leq y_j$$
$$y_j \in \{0,1\}.$$

By Lemma 4, the K-honeycomb heuristic, for $K = \alpha(n/(f-sv))^{2/3}(1+\epsilon)$ and $\epsilon \to 0$, satisfies

$$Z_K \sim 3\alpha[(f-sv)n^2]^{1/3} \quad \text{almost surely.}$$

Thus

$$Z \geq \max_{v \geq 0} 3\alpha[(f-sv)n^2]^{1/3} + nv \quad \text{almost surely.}$$

The value of v that yields the maximum is given by

$$n/s = \alpha(n/(f-sv))^{2/3}$$

This implies $n/K = s/(1+\epsilon)$. By an appropriate choice of ϵ we can show as in Lemma 6 that the K-honeycomb heuristic is almost surely feasible. Thus

$$Z \sim \max_{v \geq 0} 3\alpha[(f-sv)n^2]^{1/3} + nv \quad \text{almost surely.}$$

Therefore

$$Z \leq 3\alpha[(f-sv)n^2]^{1/3} + nv \quad \text{almost surely}$$

where v is the value defined by equation (5.6).

So $Z^I/Z \geq (3\beta[(f-sv)n^2]^{1/3}+nv)/(3\alpha[(f-sv)n^2]^{1/3}+nv) \geq \beta/\alpha$ almost surely since $\beta < \alpha$.

Finally we prove $Z_D \sim Z^I$. By (2.3), $Z_D = Z_{DC}^B$.

$$Z_{DC}^B = \min \Sigma\Sigma c_{ij}x_{ij} + \Sigma fy_j$$

$$(x,y) \in P(DC) \cap P(INT).$$

Z_{DC}^B reduces to a linear program very similar to Z^I. The only difference is the constraint

(5.7) $\Sigma y_j \geq n/s$ in Z_{DC}^B instead of $\Sigma y_j \geq n/s$ in Z^I.

Consider a K/n-uniform fractional solution.

When $\beta(n/f)^{2/3} \geq n(1+o(1))/s$, set $K \sim \beta(n/f)^{2/3}$. Then neither of the constraints (5.7) is binding. So, by Lemma 5,

$Z_F \sim Z^I \sim Z_D$ almost surely.

When $\beta(n/f)^{2/3} < n(1+o(1))/s$, then set $K = [n/s]$. Note that $K \sim n/s$ as $n/s \to \infty$.

So $Z_F \sim Z^I \sim Z_D$ almost surely by the analysis made earlier to compute Z^I when $f > s^{3/2}/3\sqrt{\pi}n$. ☐

Lemma 8

$Z^{BI} = fn/s$

$Z^{BI}/Z \leq .3333334$ almost surely for $f \leq (\alpha s)^{3/2}/\sqrt{n}$.

Proof: The constraint $\Sigma y_j \geq n/s$ is feasible for Z^{BI}. So

$$Z^{BI} = \Sigma\Sigma \ c_{ij}x_{ij} + f\Sigma y_j \geq fn/s.$$

Now consider the following solution.

$y_j = 1/s$ for all j, $x_{ii} = 1$, $x_{ij} = 0$ for $i \neq j$. It is feasible for Z^{BI} and its value is $f\ n/s$. So $Z^{BI} = f\ n/s$.

When $f \leq (\alpha s)^{3/2}/\sqrt{n}$, then $Z \sim 3\alpha(fn^2)^{1/3}$ almost surely.

$Z \geq 3(1-o(1))\ (f^2n)^{1/3}(fn^2)^{1/3}/s = 3(1-o(1))fn/s$ almost surely.

This shows $Z^{BI}/Z \leq 1/3+o(1)$ almost surely. ☐

6. Conclusion

In this paper we have studied relaxations of CPLP from three different angles: (i) inequalities among the corresponding bounds, (ii) computational experiments and (iii) probabilistic analysis. The results in (i) can be thought of as a ready reckoner to indicate the relaxation that is easiest to solve among those that yield the same bound. We also showed that relaxations based on the variable splitting method do not provide different bounds than the classical Lagrangian relaxations.

In the probabilistic study, where we assume equal fixed costs and capacities at all facilities, we showed that the relaxations fall into three categories according to their strength. The weak LP relaxation can be solved analytically but the bound is indeed very weak. The strong LP relaxation almost surely gives a bound within one fifth of one percent of the optimum. The relaxation proposed by Nauss falls into the same category. These relations are relatively easy to solve. Finally, the relaxation proposed by Van Roy is extremely tight as it almost surely provides a bound that goes to Z as the size of the problem increases. But this relaxation is strongly NP-hard to compute.

The computational results complement our theoretical results. For Z_C and Z_D we found that problems that are either loosely or tightly capacitated have smaller duality gaps compared to those in the intermediate range. Based on our computational results we suggest the use of a relaxation based heuristic (H) to solve CPLP. It provides considerably better solutions that the VSM heuristic and is also superior on computational time, at least on the large problems. Further, it is obtained in conjunction with a lower bound.

References

[1] S. Ahn, C. Cooper, G. Cornuejols and A. Frieze, "Probabilistic Analysis of a Relaxation for the K-Median Problem", Mathematics of Operations Research 13 (1988), 1-31.

[2] U. Akinc and M. Khumawala, "An Efficient Branch and Bound Algorithm for the Capacitated Warehouse Location Problem", Management Science 23 (1977), 585-594.

[3] B. M. Baker, "Linear Relaxations of the Capacitated Warehouse Location Problem", Journal of the Operational Research Society 33 (1982), 475-479.

[4] J. Barcelo, E. Fernandez and K. O. Jönsten, "Computational Results from a New Lagrangian Relaxation Algorithm for the Capacitated Plant Location Problem", working papers CMI-no862090-4, Bergen (1986).

[5] D. C. Cho, E. L. Johnson, M. W. Padberg and M. R. Rao, "On the #? Plant Location Problem I: Valid Inequalities and Facets", Mathematics of Operations Research 8 (1983), 579-589.

[6] N. Christofides and J. E. Beasley, "An Algorithm for the Capacitated Warehouse Location Problem", European Journal of Operational Research 12 (1983), 19-28.

[7]   P. S. Davis and T. L. Ray, "A Branch and Bound Algorithm for the Capacitated Facilities Location Problem", _Naval Research Logistics Quarterly_ 16 (1969), 331-343.

[8]   W. Domschke and A. Drexl, "ADD-heuristics' Starting Procedures for Capacitated Plant Location Models", _European Journal of Operational Research_ 21 (1985), 47-53.

[9]   J. Edmonds, "Paths, Trees and Flowers", _Canadian Journal of Mathematics_ 17 (1965), 449-467.

[10] D. Erlenkotter, "A Dual-based Procedure for #? Facility Location", _Operations Research_ 26 (1978), 992-1009.

[11] D. Fayard and G. Plateau, "An Algorithm for the Solution of the 0-1 Knapsack Problem", _Computing_ 28 (1982), 269-287.

[12] M. R. Garey and D. S. Johnson, _Computers and Intractability: A Guide to the Theory of NP-Completeness_, Freeman, San Francisco (1979).

[13] A. M. Geoffrion, "Lagrangian Relaxation and its Uses in Integer Programming", _Mathematical Programming Study 2_ (1974), 82-114.

[14] A. M. Geoffrion and R. McBride, "Lagrangian Relaxation Applied to Capacitated Facility Location Problems", _AIIE Transactions_ 10 (1978), 40-47.

[15] M. Grötschel, L. Lovasz and A. Schrijver, "The Ellipsoid Method and its Consequences in Combinational Optimization", Combinatorica 1 (1981), 169-197.

[16] M. Guignard and S. Kim, "A Strong Lagrangian Relaxation for Capacitated Plant Location Problems", Working Paper No. 56, Department of Statistics, University of Pennsylvania (1983).

[17] M. Guignard and S. Kim, "Lagrangean Decomposition: A Model Yielding Stronger Lagrangian Bounds", Mathematical Programming, 39 (1987), 215-228.

[18] M. Guignard and K. Spielberg, "A Direct Dual Method for the Mixed Plant Location Problem with Some Side Constraints", Mathematical Programming 17 (1979), 198-228.

[19] Hoeffding, " Probability Inequalities for Sums of Bounded Random Variables", Journal of the American Statistical Association 58 (1963), 13-30.

[20] S. K. Jacobsen, "Heuristics for the Capacitated Plant Location Model", European Journal of Operational Research 12 (1983), 253-261.

[21] B. M. Khumawala, "An Efficient Heuristic Procedure for the Capacitated Warehouse Location Problem", Naval Research Logistics Quarterly 21 (1974), 609-623.

[22] G. A. Kochman and C. J. McCallum Jr., "Facility Location Models for Planning Transatlantic Communications Network", European Journal of Operational Research 6 (1981), 205-211.

[23] A. A. Kuehn and M. J. Hamburger, "A Heuristic Program for Locating Warehouses", Management Science 9 (1963), 643-666.

[24] A. Mirzaian, "Lagrangian Relaxation for the Star-star Concentrator Location Problem: Approximation Algorithm and Bounds", Networks 15 (1985), 1-20.

[25] R. M. Nauss, "An Improved Algorithm for the Capacitated Facility Location Problem", Journal of the Operational Research Society 29 (1978), 1195-1201.

[26] C. H. Papadimitriou, "Worst-case and Probabilistic Analysis of a Geometric Location Problem", SIAM Journal on Computing 10 (1981), 542-557.

[27] G. Sa, "Branch and Bound and Approximate Solutions to the Capacitated Plant Location Problem", Operations Research 17 (1969), 1005-1016.

[28] R. Sridharan, "The Capacitated Plant Location Problem and Extensions", Ph. D. dissertation, Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh, (1986).

[29] V. Srinivasan and G. L. Thompson, "An Operator Theory of Parametric Programming for the Transportation Problem", <u>Naval Research Logistics Quarterly</u> 19 (1972), 205-252.

[30] T. J. Van Roy, "A Cross Decomposition Algorithm for Capacitated Facility Location", <u>Operations Research</u> 34 (1986), 145-163.

END

DATE

FILMED

DTIC

JULY 88